

<b>INTRODUCCIÓN AL CAPÍTULO</b>	<b>3</b>
<b>5. PLC: MICROLOGIX 1500</b>	<b>4</b>
5.1. Descripción general del hardware .....	4
5.2. Conexiones de comunicación .....	4
5.3. Entradas y salidas .....	6
5.3.1. Esquema de bloques de terminales .....	6
5.3.2. Direccionamiento .....	6
5.4. Memoria .....	7
5.4.1. Memoria de usuario .....	7
5.4.2. Protección de los archivos de datos durante la descarga .....	9
5.5. Consideraciones de seguridad .....	10
5.6. Especificaciones del controlador .....	10
5.7. Glosario básico .....	11
5.8. Bibliografía del capítulo .....	12





## Introducción al capítulo

En este capítulo se describirán los aspectos más importantes del controlador\*, incidiendo en aquellos aspectos que serán más necesarios para la realización de las prácticas a distancia. El autómatas\* programable MicroLogix 1500 pertenece a la gama de autómatas de rango medio de Allen Bradley y puede acceder a Ethernet mediante la conexión con el módulo ENI (*Ethernet Network Interface*).



Figura 5.1: MicroLogix 1500 de Allen Bradley [1]

Para ampliar información sobre el dispositivo de control se puede recurrir a las indicaciones del apartado 5.8, donde se exponen los documentos consultados para la elaboración de este capítulo.

\* Referenciados en el Glosario



## 5. PLC: MicroLogix 1500

### 5.1. Descripción general del hardware

El MicroLogix 1500 es una plataforma de control lógico programable que cuenta con un innovador diseño de dos piezas y medidas pequeñas. El procesador\* y la base (figura 5.2) se deslizan juntos para formar el controlador completo. Estos se reemplazan independientemente, lo cual permite maximizar las opciones de E/S\* incorporadas y minimizar los costos de inventario.[2]

El controlador está formado por los siguientes componentes: una fuente de alimentación, circuitos de entrada, circuitos de salida y un procesador, y está pensado para montarse sobre un carril DIN. En nuestro caso se dispone de:

- Unidad base modelo 1764-24BWA: 12 entradas a 24 V<sub>CC</sub> y 12 salidas de relé\*.
- Fuente de alimentación a 120/240 V<sub>CA</sub>
- Procesador modelo 1764-LSP, con 7Kb de capacidad para programa de usuario.

Para programar el controlador se usa el software RSLogix 500 de Rockwell Software. En el capítulo 8 del presente curso se hace referencia a los puntos más relevantes del mismo.

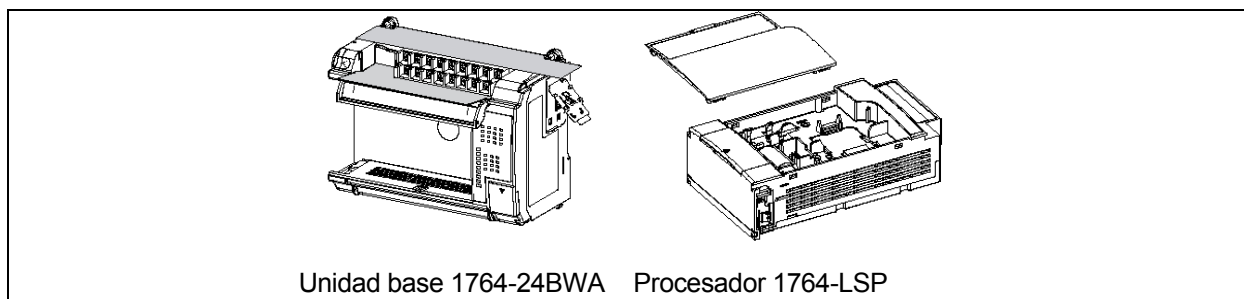


Figura 5.2: Partes del controlador MicroLogix 1500 [3]

### 5.2. Conexiones de comunicación

El puerto denominado en la documentación como Canal 0 corresponde al puerto RS-232\* del autómatas. Éste permitirá la conexión del autómatas con el puerto serie del ordenador personal para poder programarlo de manera directa, o con dispositivos de interface de red\* (ENI en nuestro caso) para los que incorpora alimentación de 24 V<sub>CC</sub>. [2]

Este puerto utiliza el protocolo\* de comunicación DF1 Full-duplex\*, muy útil cuando se requiere comunicación RS-232 punto a punto. Este protocolo acepta transmisiones



\* Referenciados en el Glosario

simultáneas entre dos dispositivos en ambas direcciones. El protocolo DF1 controla el flujo de mensajes, detecta y señala errores y efectúa reintentos si se detectan errores.[4] Presenta por defecto las siguientes características:

Parámetro	Opciones	El valor predeterminado
Baud Rate	300, 600, 1200, 2400, 4800, 9600, 19.2 K, 38.4 K	19.2 K
Parity	ninguna, par	Ninguno
Source ID (Dirección de nodo)	0 a 254 decimal	1
Control Line	sin handshaking, comunicación de módem Full-Duplex	sin handshaking
Error detection	CRC, BCC	CRC
Embedded Responses	detección automática, habilitada	Auto detect
Duplicate Packet (Message) Detect	habilitada, inhabilitada	Activado
Timeout ACK	1 a 65535 conteos (incrementos de 20 ms)	50 veces
Nak retries	0 a 255	3 intentos
ENQ retries	0 a 255	3 intentos
Stop bits	no es un parámetro, siempre es 1	1

Figura 5.3: Parámetros de configuración para Full-duplex DF1 [3]

Como se puede ver en la figura 5.3 la velocidad de transmisión\* por defecto es de 19200 Bps. Este dato deberá ser tomado en cuenta más tarde en la configuración del módulo ENI. Para que no existan problemas de comunicaciones, los dos dispositivos deberán tener la misma velocidad de transmisión. Este puerto puede también configurarse mediante el software RSLogix 500 como se muestra en la figura 5.4, accediendo al menú *Channel Configuration* del árbol de proyecto.

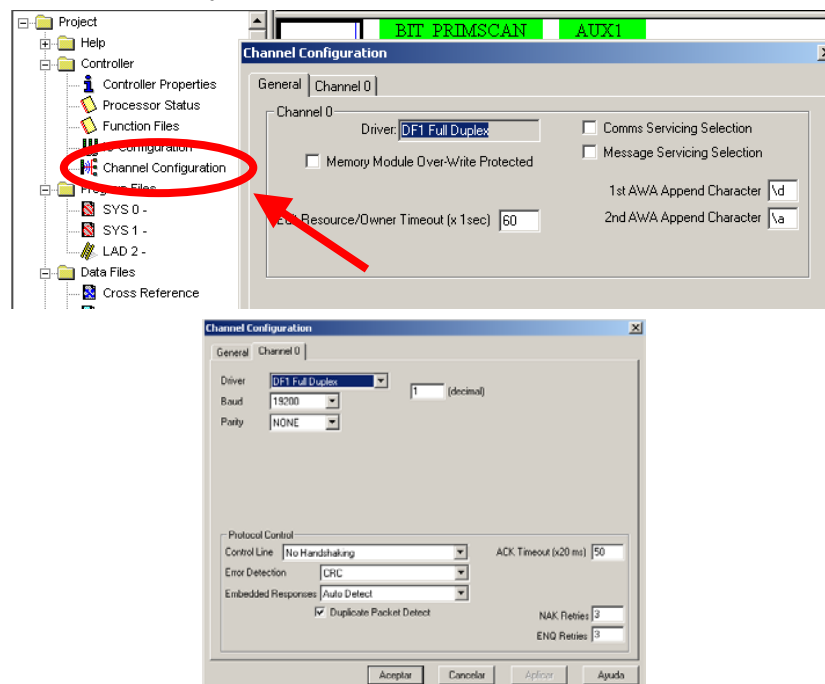


Figura 5.4: Menú de configuración del Canal 0

\* Referenciados en el Glosario



## 5.3. Entradas y salidas

### 5.3.1. Esquema de bloques de terminales

El controlador utilizado contiene únicamente el bloque de entradas/salidas\* incorporadas, es decir, 12 entradas a 24 V<sub>CC</sub> y 12 salidas de relé. Debido a la tipología de las prácticas donde no se emplean dispositivos de entrada (pulsadores, sensores,...) sino que se simulan mediante software, las entradas no se usarán. Las salidas al ser de tipo relé deberán cablearse a un potencial de referencia, en nuestro caso se utilizará el potencial de +24 V<sub>CC</sub> proporcionado por la propia fuente de alimentación del autómat. En la figura 5.5 se muestra el cableado de uno de estos potenciales que corresponde a la salida O:0/0, en rojo la alimentación y en negro el común.

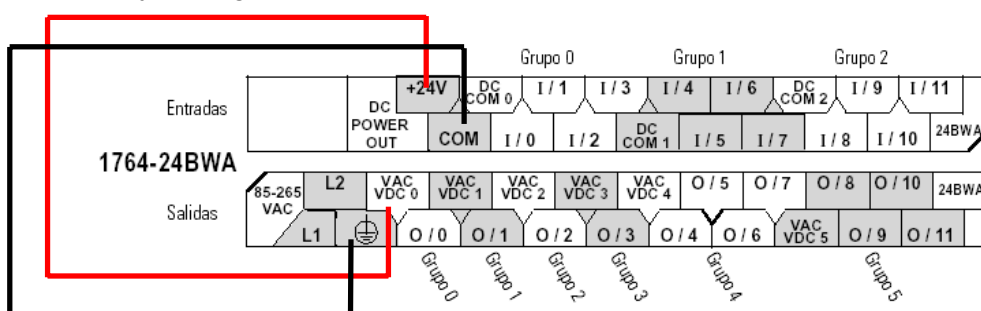


Figura 5.5: Terminales de la unidad base [3]

Los terminales de la unidad base están agrupados diferenciándose en cinco grupos de salidas (según el sombreado, figura 5.5), de esta manera sería posible cablear las salidas a potenciales diferentes. En nuestro caso todas se encuentran cableadas a 24 V<sub>CC</sub>.

### 5.3.2. Direccionamiento

Los estados de las entradas y salidas del autómat se pueden verificar mediante el RSLogix, accediendo al menú *Data Files>Output* o *Input*. En este menú se puede visualizar su estado, forzar bits\* y nombrar las salidas. El autómat se refiere a las diferentes entradas y salidas digitales, las que se tienen en este caso, como [4]:

Tipo de archivo*	Delimitador de ranura	Nº ranura	Delimitador de bit	Nº de ranura
I (entrada)	:	0	/	0 a 11
O (salida)	:	0	/	0 a 11

Figura 5.6: Nomenclatura entradas y salidas

Por ejemplo la salida número 2 se escribiría: O:0/2.

Para el direccionamiento de las variables internas de la memoria del autómat se usa una nomenclatura similar. La diferencia es que en lugar de llamarse I ó O, se llaman B3. Por



\* Referenciados en el Glosario

ejemplo la primera variable corresponde a B3:0/0. Éstas se reúnen en grupos de 16 variables. La número 17 sería B3:1/0. Y así de manera sucesiva hasta que lo permita la memoria de usuario que posea nuestro controlador.

El forzado de variables nos será muy útil para simular los pulsadores de paro, marcha, etc. Este forzado se puede realizar con el autómata *on-line* (en línea) y *offline* (fuera de conexión). Para forzar una variable de bit se debe entrar el menú *Data Files> Binary*, y pasar el bit correspondiente (círculo azul) de 0 a 1 como se muestra en la figura 5.7. En este caso estaríamos activando la variable B3:1/1.

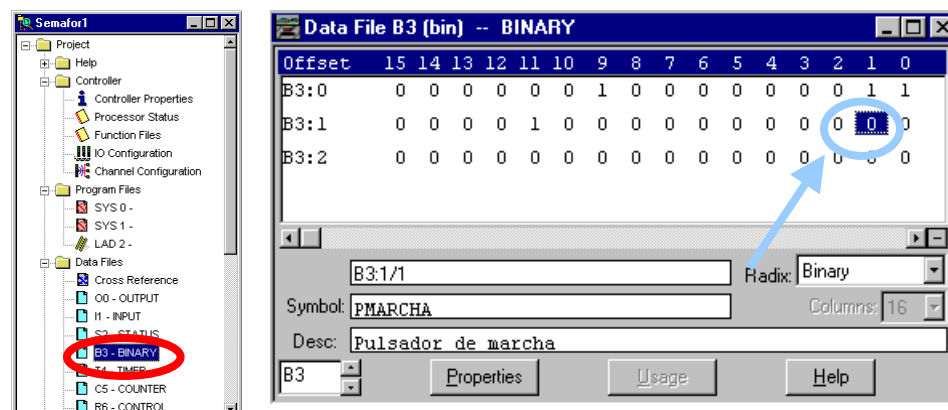


Figura 5.7: Menú *Binary*

Vemos como es posible nombrar las variables utilizadas (*Symbol*, figura 5.7) y dotarlas de una descripción en este mismo menú (*Desc.*, figura 5.7). Esto será útil para la creación del programa ya que las variables estarán identificadas y no se producirán confusiones.

## 5.4. Memoria

### 5.4.1. Memoria de usuario

La memoria de usuario es la cantidad de almacenamiento disponible de un usuario para almacenar lógica de escalera, archivos de tabla de datos, configuración de E/S, etc., en el controlador. Contiene archivo de datos, de programa y de función (figura 5.8). [4]

En la memoria de nuestro controlador se encuentran los siguientes archivos:

Archivos de datos		Archivos de función		Archivos de programa		Archivos especiales	
0	Archivo de salida	HSC	Contador de alta velocidad	0	Archivo sistema 0	0	Archivos de recetas 0
1	Archivo de entrada	STI	Interrupción temporizada seleccionable	1	Archivo sistema 1	1	Archivos de recetas 1
2	Archivo de estado	EII	Interrupción de entrada de evento	2	Archivo programa 2	2 a 255	Archivos de recetas 2 a 255
3	Archivo de bits	RTC	Reloj en tiempo real	3 a	Archivo programa 3		



4	Archivo de temporizador	TPI	Información del potenciómetro de ajuste	255	a 255		
5	Archivo de contador	MMI	Información del módulo de memoria				
6	Archivo de control	DAT	Herram. de acceso a datos				
7	Archivo de enteros	BHI	Información de hardware base				
8	Archivo de punto flotante	CS	Estado de comunicaciones				
9 a 255	(B) Bit (T) Temporizador (C) Contador (R) Control (N) Enteros...	IOS	Estado de E/S				

Figura 5.8: Distribución de archivos de la memoria de usuario [4]

Los archivos de datos del usuario consisten en archivos de estado del sistema: archivos de imagen de E/S y todos los archivos de datos que puede crear el usuario (bit, temporizador, contador, control, enteros, cadena, palabra larga, MSG y PID). Una palabra de usuario se define como una unidad de memoria en el controlador, el consumo de memoria se realiza de la siguiente manera. [4]

- Archivos de datos: la palabra de usuario equivale a 16 bits de memoria. Por ejemplo una palabra larga equivale a dos palabras de usuario y un temporizador a tres. En nuestro caso solo se utilizarán variables de datos de 1 bit.
- Archivos de programa: una palabra es equivalente a una instrucción\* de lógica de escalera con un operando. Por ejemplo una instrucción XIC tiene un operando y consume una palabra, la instrucción ADD tiene tres operandos y por tanto consume tres palabras (ver instrucciones de lógica escalera en capítulo 8).
- Archivos de función: no consumen memoria de usuario.

El procesador utilizado cuenta con una memoria de 7Kb (figura 5.9). La memoria puede utilizarse para archivos de programa y archivos de datos (máximo 4Kb palabras). [4]



Figura 5.9: Memoria del procesador 1764-LSP [4]

Para verificar la fracción de memoria que está siendo utilizada en el momento de la programación se pueden seguir los siguientes pasos (figura 5.10):





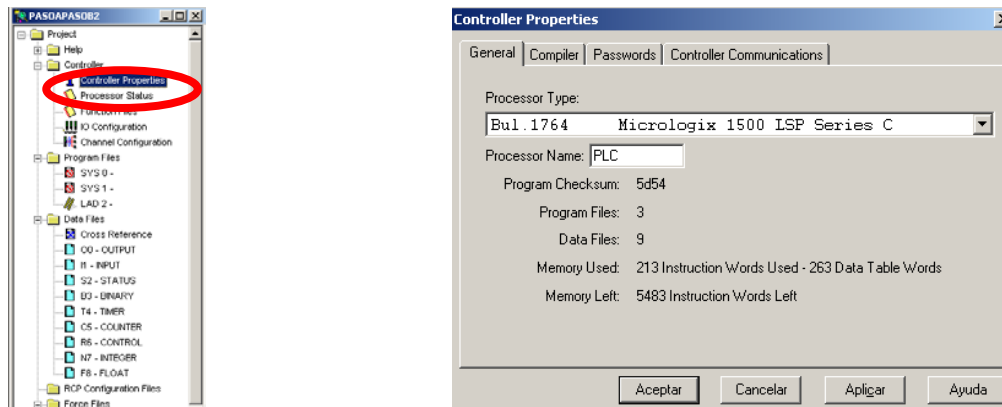


Figura 5.10: Propiedades del controlador

Vemos como se indica la cantidad de memoria utilizada y el que tipo de archivos que la ocupan (*Memory used*) y la memoria restante (*Memory left*).

#### 5.4.2. Protección de los archivos de datos durante la descarga

Puede ser necesaria cuando una aplicación\* necesita ser actualizada, pero los datos relevantes a la instalación necesitan permanecer intactos [4]. En el caso presente, en el que se simulan los pulsadores con el forzado de variables de tipo bit, es importante que el archivo de datos Binario (B) esté completamente a cero. De esta manera se evitará que el pulsador de marcha esté activado con antelación y provoque el mal funcionamiento de la planta. Así, en la primera descarga\* de programa al autómatas se deberá tener todas las variables de tipo bit utilizadas a cero y activar el *Memory Module/Download* (flecha inferior roja, figura 5.11).

En este caso lo hemos hecho para el archivo de datos Binario pero se podría hacer para otros archivos: salida (O), entrada (I), temporizador (T), contador (C), control (R), enteros (N), punto flotante (F), cadena (ST), palabra larga (L), derivada proporcional integral (PD), mensaje (MG) y interruptor de final de carrera programable (PLS).

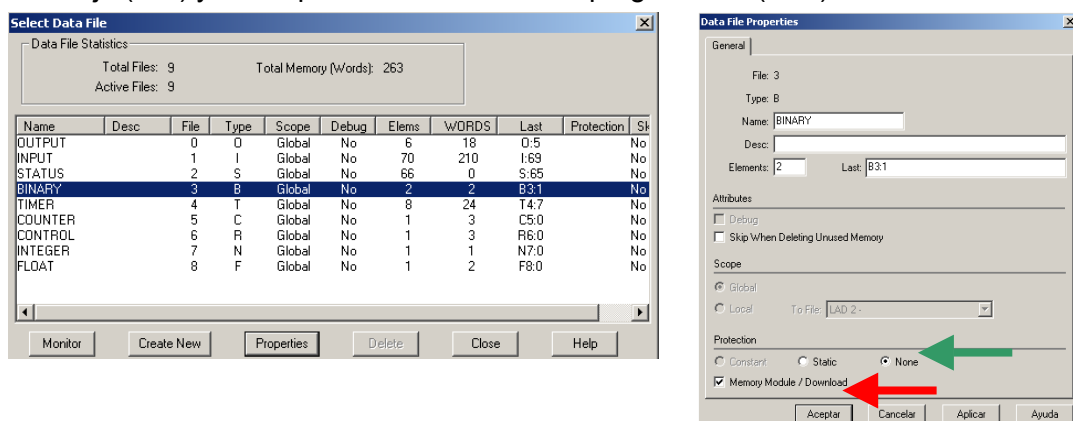


Figura 5.11: Protección de archivos

\* Referenciados en el Glosario




La protección de los archivos de datos solo se producirá si el número, tipo y tamaño de archivos de datos que contiene el controlador es exactamente igual que el que se está transmitiendo. Si alguno de estos requisitos no se cumple, todo el programa de usuario se transmite al controlador incluidos los archivos de datos. Los archivos de datos pueden estar bajo protección estática, en la que los valores contenidos en éste no pueden cambiarse a través de las comunicaciones, excepto durante una descarga de programa al controlador (flecha superior verde, figura 5.11).[4]

Estas configuraciones de protección se realizan a través del software de programación RSLogix 500.

## 5.5. Consideraciones de seguridad

Las consideraciones de seguridad son un elemento importante para la instalación apropiada del sistema.[3] En el caso actual se han tenido en cuenta que:

- Todos los cables conectados al autómatas deben permanecer en un envoltorio o ser protegidos por conductos u otra manera de protección. En este caso se ha optado por un envoltorio en espiral (figura 5.12 adyacente).
- 
- Transformador de aislamiento. Para la protección de la línea de corriente alterna que va al controlador se ha utilizado un transformador de aislamiento. Este tipo de transformador proporciona aislamiento desde el sistema de distribución y reduce el ruido eléctrico que entra al controlador. En nuestro caso se ha usado un transformador monofásico Minaven modelo MO001 16830 5R5 [5]
  - Para la mayoría de las aplicaciones, el enfriamiento por convección normal mantiene el controlador dentro del rango de temperatura especificado. Pero es recomendable mantener un espacio libre adecuado entre los componentes.

## 5.6. Especificaciones del controlador

A continuación se exponen las especificaciones más importantes para la comprensión de los diferentes componentes que forman el controlador MicroLogix 1500:

Descripción	1764-24BWA
Número de E/S	12 entradas y 12 salidas
Alimentación de la línea	85 a 265 V <sub>CA</sub> a 47 a 63 Hz
Salida de la alimentación de usuario	24 V <sub>CC</sub> a 400 mA, 400 µF máx.



Tipo de circuito de salida	Relé
Temperatura de operación	+0°C a +55°C ambiente
Humedad de operación	5% a 95% de humedad relativa (sin condensación)
Software de programación	RSLogix 500 versión 4.00.00 o posterior

Figura 5.14: Especificaciones generales [3]

Las salidas de relé tienen las siguientes características para el voltaje aportado:

Tensión ( $V_{cc}$ )	I cierre/apertura (A)	I continua (A)	Capacidad nominal cierre/apertura (VA)
24	1.2	2.0	28

Figura 5.15: Tabla de capacidades nominales de contactos de relé [3]

## 5.7. Glosario básico

<b>aplicación</b>	1) Una máquina o proceso monitoreado y controlador por un controlador. 2) El uso de rutinas basadas en computadora o procesador para fines específicos. [4]
<b>archivo</b>	Una recolección de datos o lógica organizada en grupos. [4]
<b>autómata</b>	Equipo electrónico, programable en lenguaje no informático, diseñado para controlar en tiempo real y en ambiente de tipo industrial, procesos secuenciales. [4]
<b>bit</b>	La unidad de memoria más pequeña usada en la lógica discreta o binaria, donde el valor 1 representa activado y el valor 0 representa desactivado. [4]
<b>carga</b>	La transferencia de datos desde el controlador a un dispositivo de programación o almacenamiento. [4]
<b>controlador</b>	Un dispositivo, tal como un controlador programable, usado para controlar dispositivos de salida. [4]
<b>descarga</b>	La transferencia de archivos de datos o programas a un dispositivo. [4]
<b>full-duplex</b>	Un modo de comunicación donde los datos pueden transmitirse y recibirse simultáneamente (a diferencia de half-duplex). [4]
<b>instrucción</b>	Un mnemónico que define una operación que va a ser realizada por el procesador. Un renglón en un programa consta de un conjunto de instrucciones de entrada y salida. Las instrucciones de entrada son evaluadas por el controlador como verdaderas o falsas. A su vez, el controlador establece las instrucciones de salida como verdaderas o



	falsas. [4]
<b>procesador</b>	La sección de toma de decisiones y almacenamiento de datos de un controlador programable. [4]
<b>protocolo</b>	Las reglas de intercambio de datos mediante comunicaciones. [4]
<b>red</b>	Una serie de estaciones (nodos) conectados por algún tipo de medio de comunicación. Una red puede tener un vínculo o muchos vínculos. [4]
<b>relé</b>	Un dispositivo que funciona eléctricamente y que conmuta mecánicamente los circuitos eléctricos. [4]
<b>RS-232</b>	Un estándar EIA (Electronics Industries Association) que especifica características eléctricas, mecánicas y funcionales para circuitos de comunicación binaria en serie. [4]
<b>velocidad en baudios o baud rate</b>	La velocidad de comunicación entre dispositivos. La velocidad en baudios generalmente se muestra en K baudios. Por ejemplo, 19.2 K baudios = 19,200 bits por segundo. [4]

## 5.8. Bibliografía del capítulo

- [1] ALLEN BRADLEY HOME PAGE.[ <http://www.ab.com/plclogic/micrologix/>, 19 de octubre de 2004].
- [2] ALLEN BRADLEY HOME PAGE. Controladores Programables MicroLogix 1500, Datos técnicos. Número de publicación 1764-TD001A-ES-P. [\[http://www.ab.com/manuals/es/cp/1761/](http://www.ab.com/manuals/es/cp/1761/), 19 de octubre de 2004].
- [3] ALLEN BRADLEY HOME PAGE. Controladores Programables MicroLogix 1500, Manual de usuario. Número de publicación 1764-UM001A-ES-P. [\[http://www.ab.com/manuals/es/cp/1761/](http://www.ab.com/manuals/es/cp/1761/), 19 de octubre de 2004].
- [4] ALLEN BRADLEY HOME PAGE. Controladores Programables MicroLogix 1200 y MicroLogix 1500, Manual de referencia del conjunto de instrucciones. Número de publicación 1762-RM001D-ES-P. [\[http://www.ab.com/manuals/es/cp/1761/](http://www.ab.com/manuals/es/cp/1761/), 19 de octubre de 2004].
- [5] MIMAVEN ELECTRICA, S.A. Transformadores monofásicos especiales serie r. [\[http://www.mimaven.com/serier.pdf\]](http://www.mimaven.com/serier.pdf), 19 de octubre de 2004].

